**LAPORAN PRAKTIKUM**

**DASAR PEMROGRAMAN**

**PERTEMUAN 9**

**ARRAY 1**

|  |  |
| --- | --- |
| **HIKMAH ALDRIN ABDILLAH** | **NIM: 2341720049** |

****

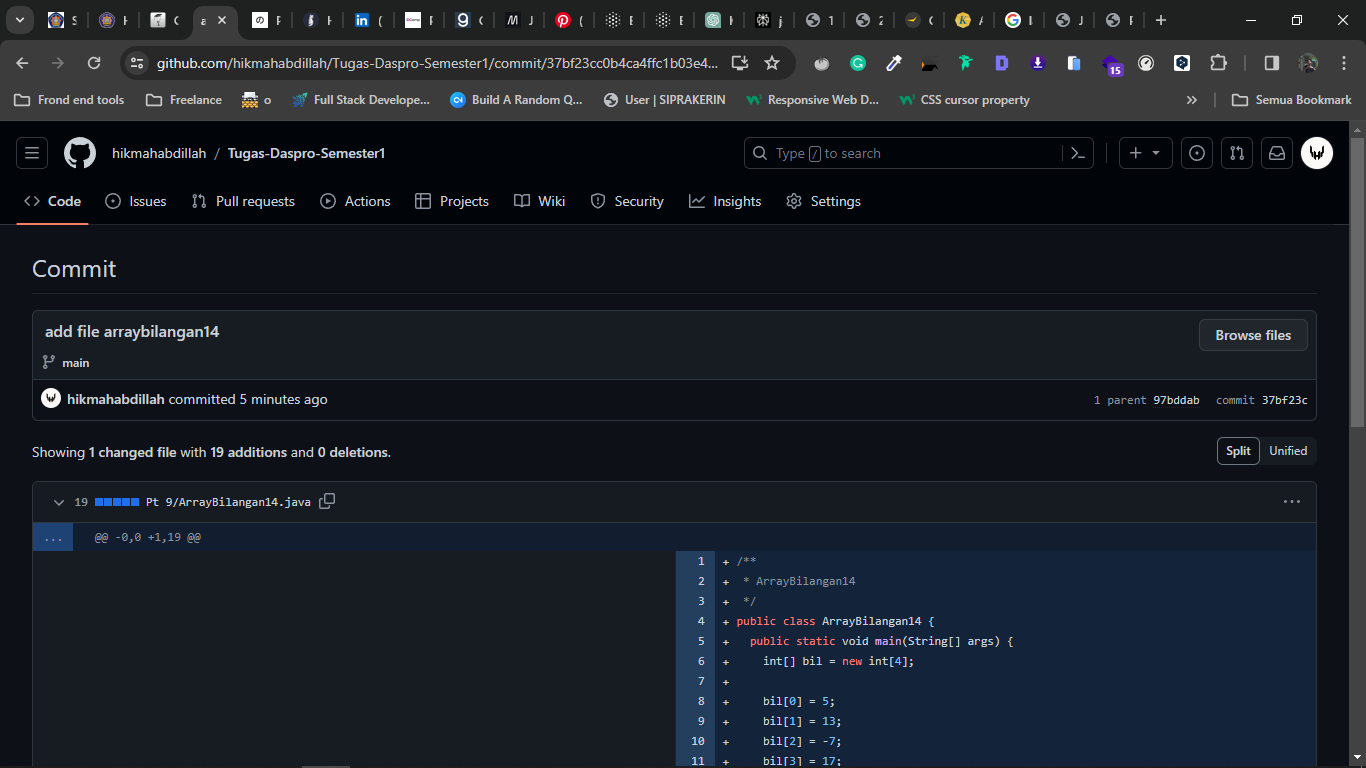
**PROGRAM STUDI D-IV TEKNIK INFORMATIKA**

**JURUSAN TEKNOLOGI INFORMASI**

**POLITEKNIK NEGERI MALANG**

**OKTOBER 2023**

**2.1 Percobaan 1: Mengisi Elemen Array**



**Pertanyaan**

1. Jika isi masing-masing elemen array bil diubah dengan angka 5.0, 12867, 7.5, 2000000.

Apa yang terjadi? Mengapa bisa demikian?

**Jawab:**

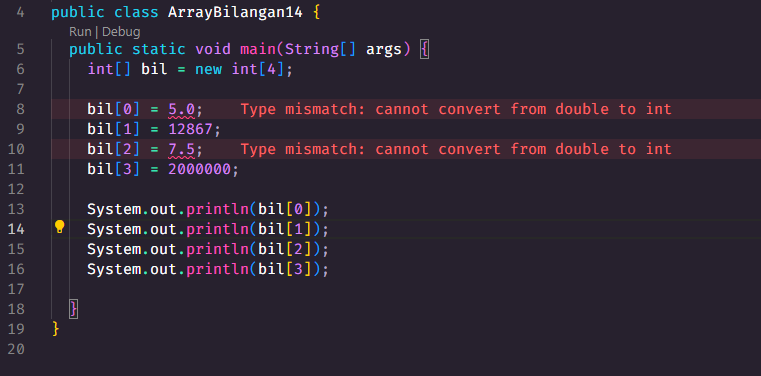
Untuk nilai dari angka 5.0 dan 7.5 terjadi error, karena array hanya dapat memuat 1 tipe data saja. Jadi pastikan untuk nilai yang ada pada dalam array sesuai dengan tipedata nya.

2. Modifikasi kode program di atas dengan melakukan inisialisasi elemen array sekaligus

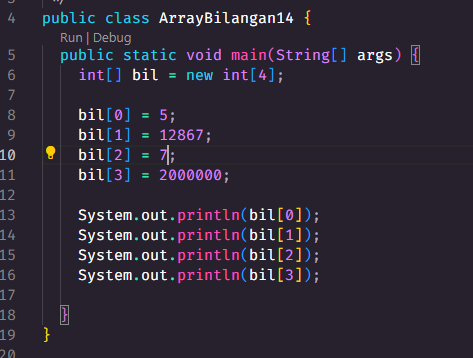
pada saat deklarasi array.

**Jawab:**

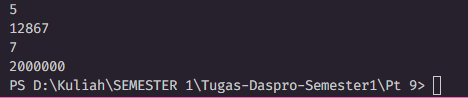
**Code yang salah**



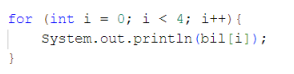
Code yang benar



Output:



3. Ubah statement pada langkah No 4 menjadi seperti berikut



Apa keluaran dari program? Jelaskan maksud dari statement tersebut.

**Jawab :**

Keluaran dari program sama saja ketika kita menggunakan sout(bil[0]) dst… , yang membedakan adalah code diatas menggunakan perulangan yang memudahkan dalam menampilkan data yang ada di dalam array tanpa harus menuliskannya satu satu.

Jadi maksud dari statement di atas adalah, melakukan perulangan sebanyak 4 kali yang dimulai dari index 0. Dimana index ke 0 ini adalah index default dari sebuah array.

Yang kemudian jika nilai kondisi yang ada di dalam for bernilai benar, maka akan menjalankan perintah sout(bil[i]). Yang dimana i ini adalah nilai index dari arraynya.

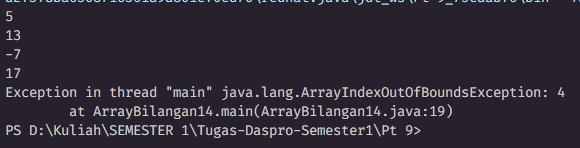
Jadi program akan menjalankan perulangan sebanyak 4 kali dengan menampilkan nilai yang ada pada array.

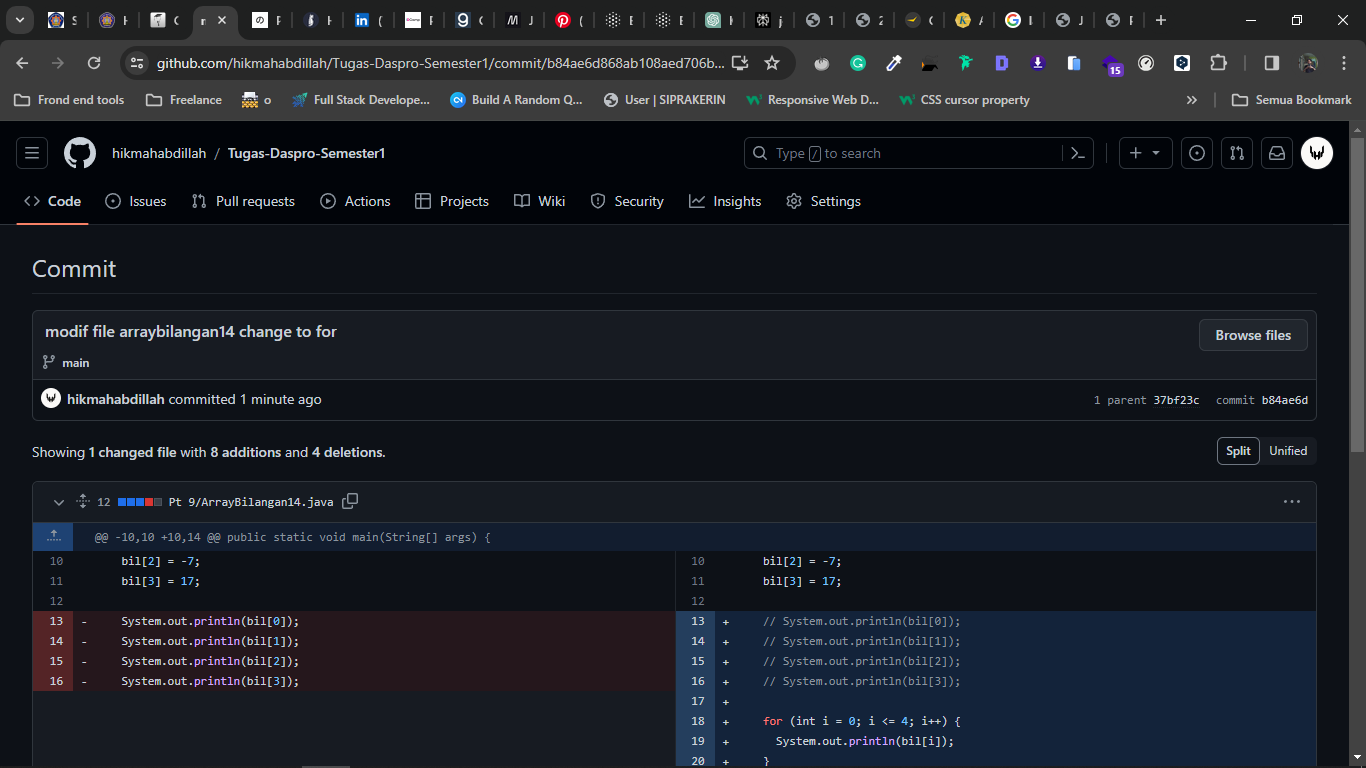
4. Jika kondisi pada statement for-loop di atas diubah menjadi: i <= 4, apa keluaran

dari program? Mengapa demikian?

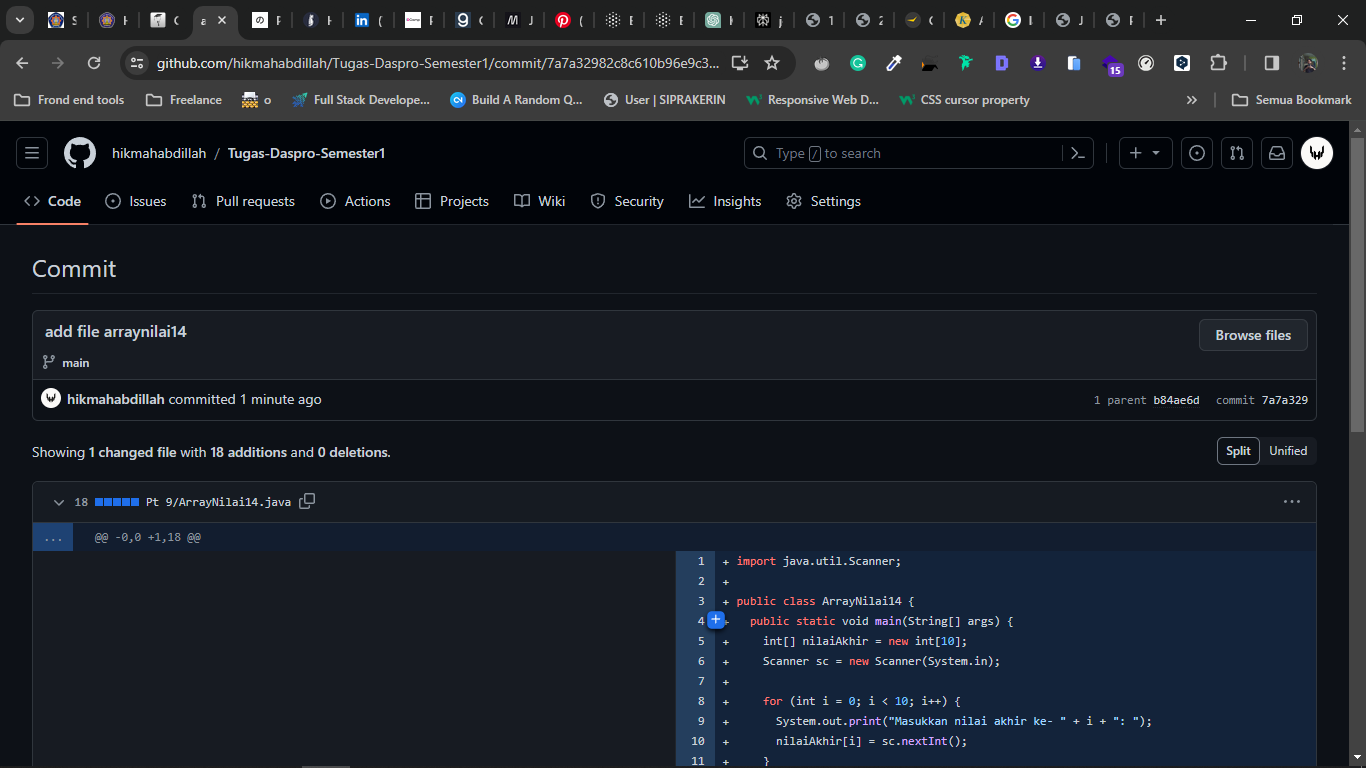
**Jawab:**

Program akan menampilkan semua data yang ada pada array seperti halnya, namun untuk index ke 4 tidak dapat ditampilkan. Karena nilai dari index ke 4 ini adalah null atau tidak ada.



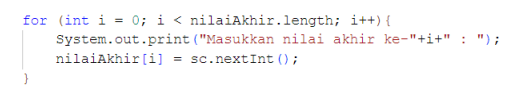
5. Push dan commit kode program ke github.

**2.2 Percobaan 2: Meminta Inputan Pengguna untuk Mengisi Elemen Array**



**Pertanyaan :**

1. Ubah statement pada langkah nomor 5 menjadi seperti berikut ini:



Jalankan program. Apakah terjadi perubahan? Mengapa demikian?

**Jawab:**

Tidak terjadi perubahan, seperti halnya dengan menggunakan for pada umumnya. Yang membedakan adalah pada penulisan kondisinya, spada bagian operasi perbandingan.

Yaitu pada syntax (i < nilaiAkhir.length) yang akhirnya I adalah index yang akan dijalankan itu kurang dari panjang nilai(jumlah elemen) dari array nilaiAkhir.

2. Apa yang dimaksud dengan kondisi: i < nilaiAkhir.length ?

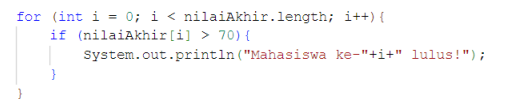
**Jawab:**

Jika nilai **i** kurang dari panjang (jumlah elemen) dari array **nilaiAkhir**.

3. Ubah statement pada langkah nomor 6 menjadi seperti berikut ini, sehingga program

hanya menampilkan nilai Mahasiswa yang lulus saja (yaitu mahasiswa yang memiliki nilai

> 70):



Jalankan program dan jelaskan alur program!

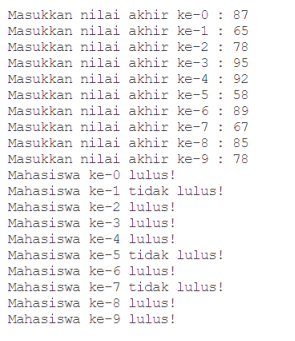
**Jawab :**

1. Program akan melakukan perulangan sebanyak elemen yang ada di array nilai Akhir
2. Kemudian di dalamnya terdapat pemilihan
3. Jika nilaiAkhir dari elemen ke-[i] dan nilai dari elemen tersebut lebih dari 70, maka print(“Mahasiswa ke-“+i+” lulus!”)

4. Modifikasi program agar menampilkan status kelulusan semua mahasiswa berdasarkan

nilai, yaitu dengan menampilkan status mana mahasiswa yang lulus dan tidak lulus,

seperti ilustrasi output berikut:



**Jawab :**

import java.util.Scanner;

public class ArrayNilai14 {

  public static void main(String[] args) {

    int[] nilaiAkhir = new int[10];

    Scanner sc = new Scanner(System.in);

    for (int i = 0; i < nilaiAkhir.length; i++) {

      System.out.print("Masukkan nilai akhir ke- " + (i + 1) + ": ");

      nilaiAkhir[i] = sc.nextInt();

    }

    // modifikasi no 4

    for (int i = 0; i < nilaiAkhir.length; i++) {

      if (nilaiAkhir[i] > 70) {

        System.out.println("Mahasiswa ke-" + (i + 1) + " lulus!");

      } else {

        System.out.println("Mahasiswa ke-" + (i + 1) + " tidak lulus!");

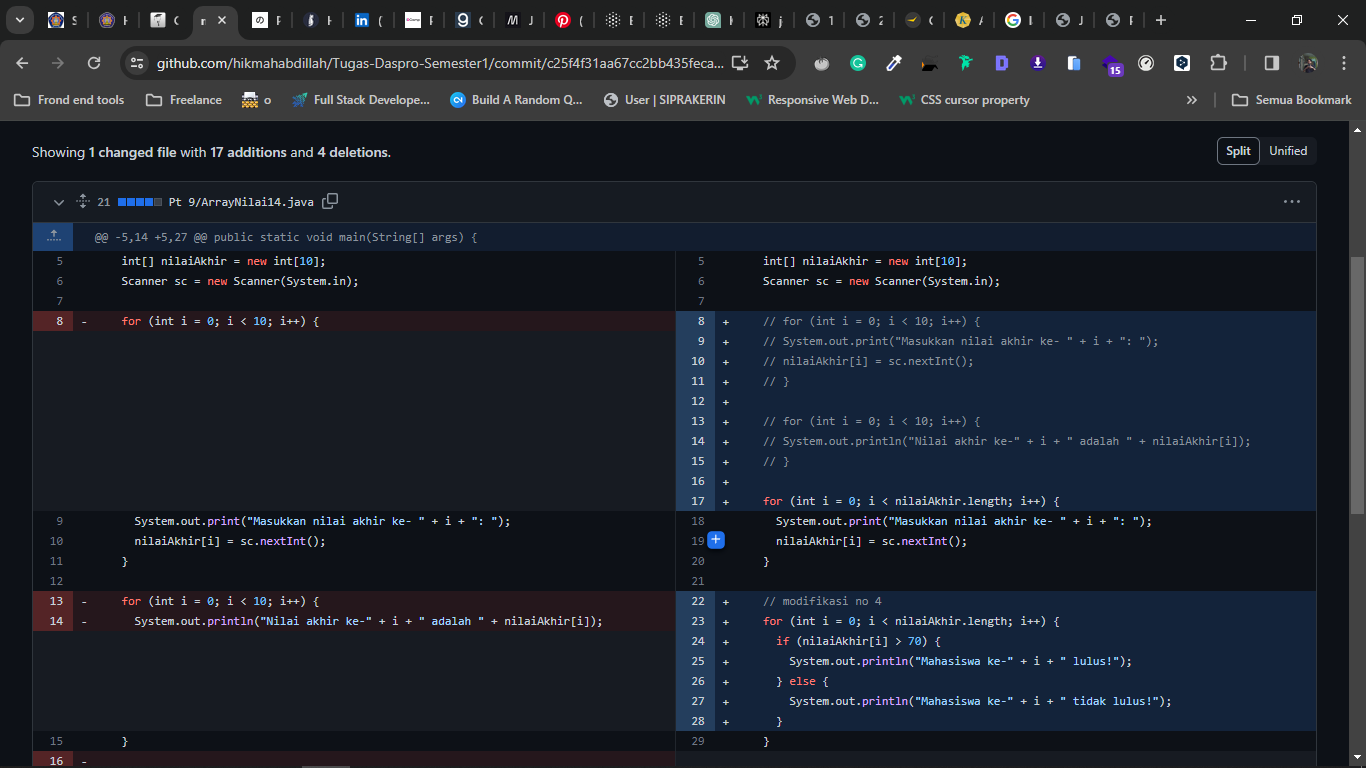
      }

    }

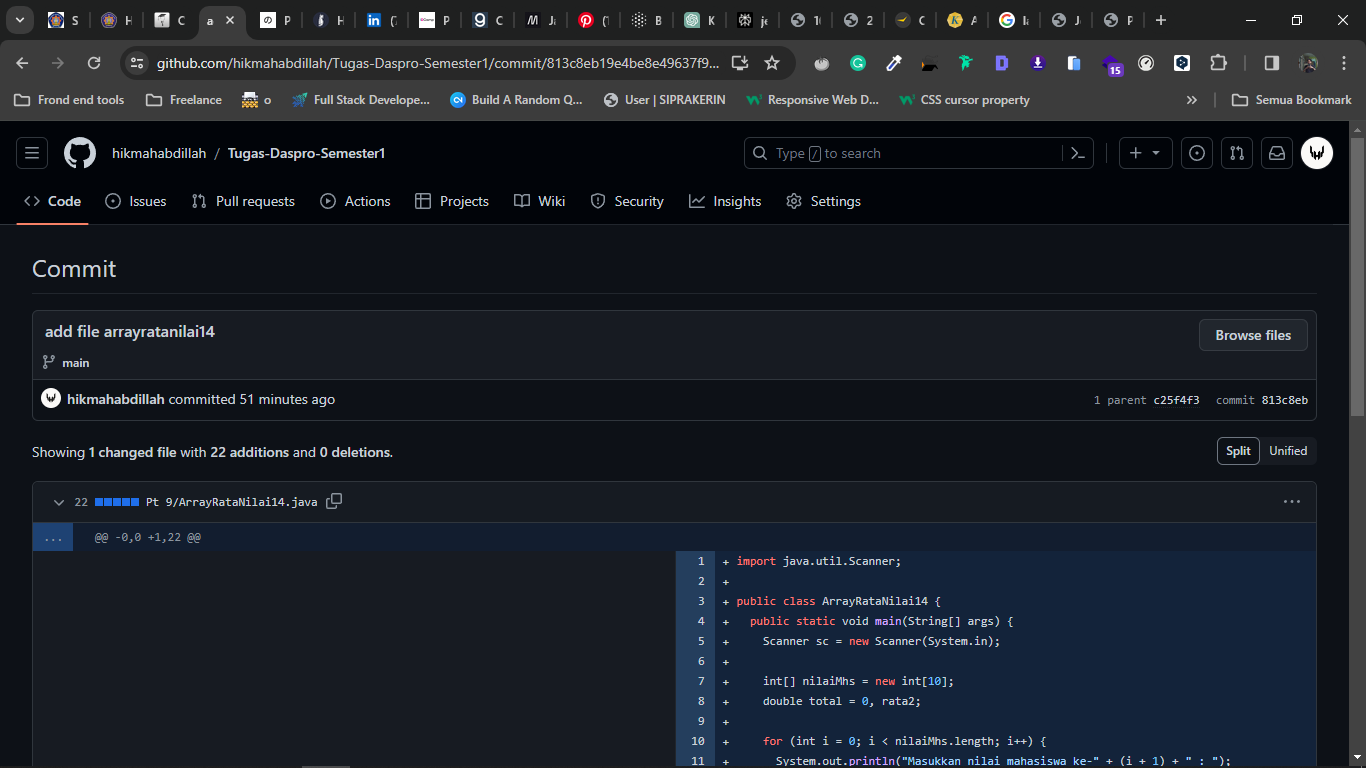
  }

}

5. Push dan commit kode program ke github.



**2.3 Percobaan 3: Melakukan Operasi Aritmatika terhadap Elemen Array**



**Pertanyaan**

1. Modifikasi kode program pada praktikum percobaan 3 di atas (ArrayRataNilaiXX.java)

agar program dapat menampilkan banyaknya mahasiswa yang lulus, yaitu mahasiswa

yang memiliki lebih besar dari 70 (>70).

**Jawab:**

import java.util.Scanner;

public class ArrayRataNilai14 {

  public static void main(String[] args) {

    Scanner sc = new Scanner(System.in);

    int[] nilaiMhs = new int[10];

    double total = 0, rata2;

    int lulus = 0;

    for (int i = 0; i < nilaiMhs.length; i++) {

      System.out.println("Masukkan nilai mahasiswa ke-" + (i + 1) + " : ");

      nilaiMhs[i] = sc.nextInt();

    }

    for (int i = 0; i < nilaiMhs.length; i++) {

      total += nilaiMhs[i];

      if (nilaiMhs[i] > 70) {

        lulus += 1;

      }

    }

    rata2 = total / nilaiMhs.length;

    System.out.println("Rata - Rata Nilai - " + rata2);

    System.out.println("banyaknya mahasiswa yang lulus : " + lulus);

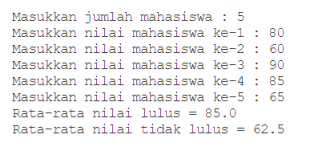
  }

}

2. Modifikasi program pada praktikum percobaan 3 di atas (ArrayRataNilaiXX.java)sehingga

program menerima jumlah elemen berdasarkan input dari pengguna dan mengeluarkan

output seperti berikut ini:



import java.util.Scanner;

public class ArrayRataNilai14 {

  public static void main(String[] args) {

    Scanner sc = new Scanner(System.in);

    int x = 0, lulus = 0, notLulus = 0;

    double rataLulus = 0, rataNotLulus = 0;

    int jmlLulus = 0, jmlNotLulus = 0;

    System.out.print("Masukkan jumlah mahasiswa : ");

    x = sc.nextInt();

    double[] nilaiMhs = new double[x];

    for (int i = 0; i < nilaiMhs.length; i++) {

      System.out.print("Masukkan nilai mahasiswa ke-" + (i + 1) + " : ");

      nilaiMhs[i] = sc.nextInt();

    }

    for (int i = 0; i < nilaiMhs.length; i++) {

      if (nilaiMhs[i] > 70) {

        lulus += nilaiMhs[i];

        jmlLulus += 1;

        rataLulus = (double) lulus / jmlLulus;

      } else {

        notLulus += nilaiMhs[i];

        jmlNotLulus += 1;

        rataNotLulus = (double) notLulus / jmlNotLulus;

      }

    }

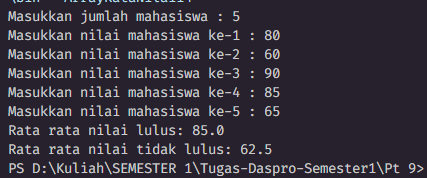
    System.out.println("Rata rata nilai lulus: " + rataLulus);

    System.out.println("Rata rata nilai tidak lulus: " + rataNotLulus);

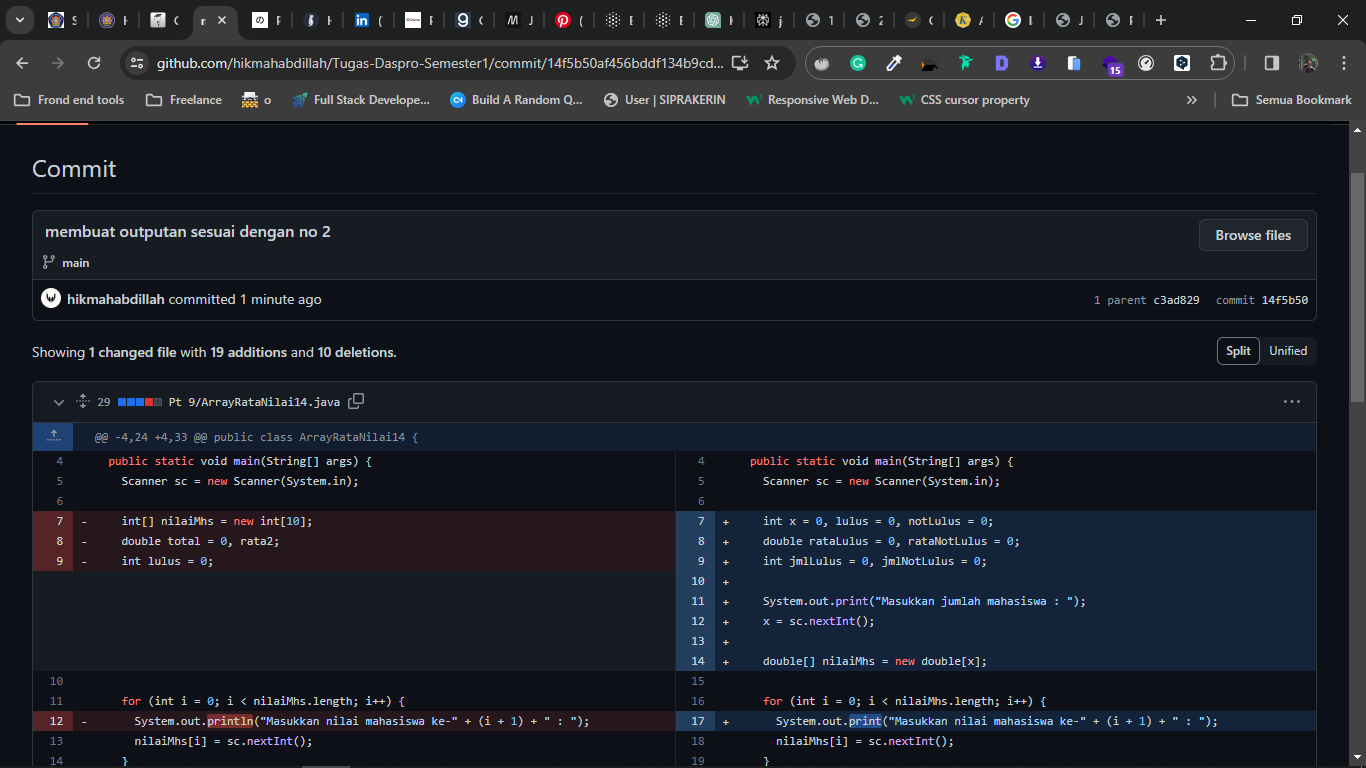
  }

}

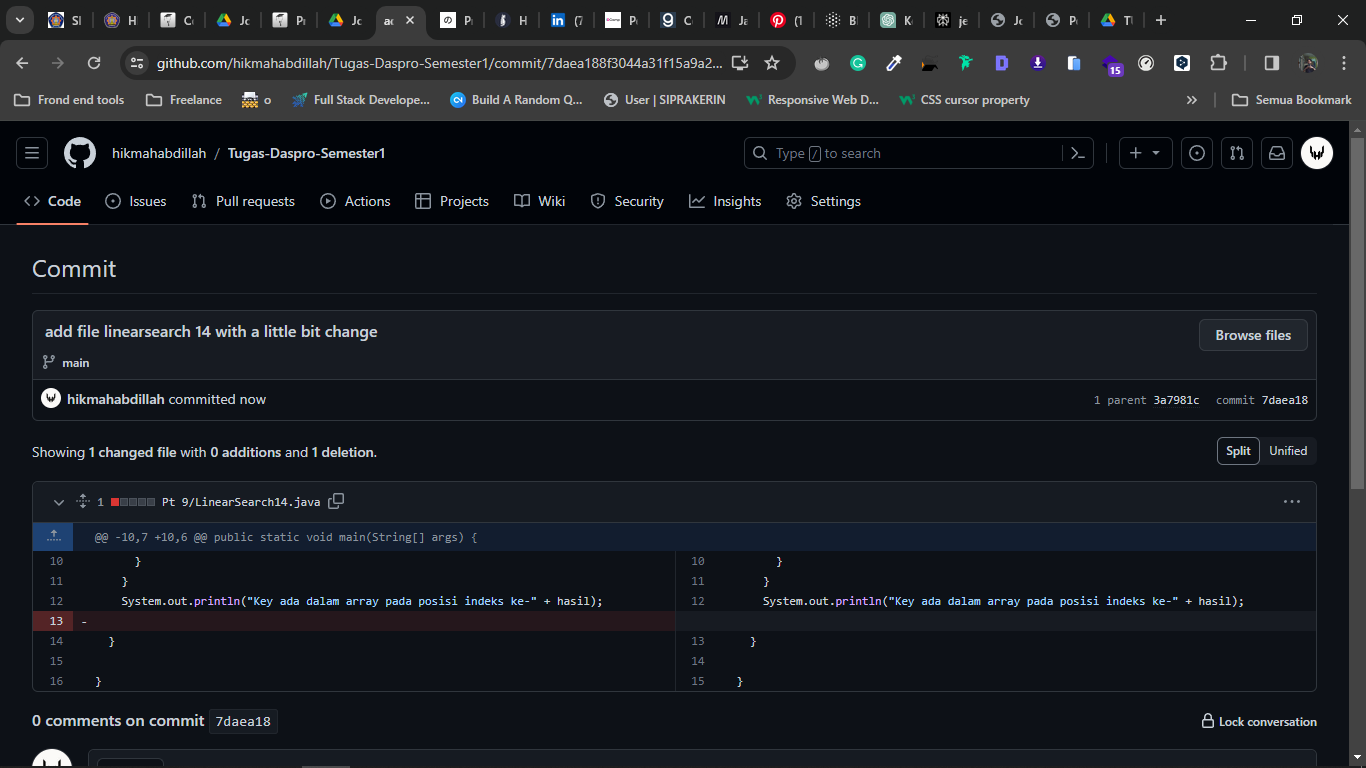
OUTPUT:



3. Push dan commit kode program ke github.



**2.4 Percobaan 4: Searching**



Pertanyaan

1. Jelaskan maksud dari statement break; pada baris ke-11 kode program percobaan 4 di

atas.

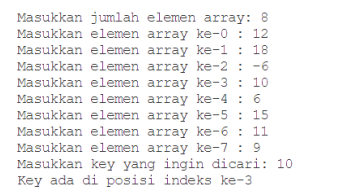
**Jawab :**

break digunakan untuk menghentikan iterasi. Hal ini dilakukan karena pencarian linear akan berhenti setelah menemukan elemen yang dicari. Tidak perlu lagi melanjutkan pencarian ke elemen-elemen selanjutnya karena kita telah menemukan elemen yang dicari.

2. Modifikasi kode program pada percobaan 4 di atas sehingga program dapat menerima

input berupa banyaknya elemen array, isi array, dan key yang ingin dicari. Lalu cetak ke

layar indeks posisi elemen dari key yang dicari. Contoh hasil program:



**Jawab:**

import java.util.Scanner;

public class LinearSearch14 {

  public static void main(String[] args) {

    Scanner sc = new Scanner(System.in);

    int key = 0;

    int hasil = 0;

    int n\_elemen = 0;

    System.out.print("Masukkan jumlah elemen array : ");

    n\_elemen = sc.nextInt();

    int[] arrayInt = new int[n\_elemen];

    for (int i = 0; i < arrayInt.length; i++) {

      System.out.print("Masukkan elemen array ke -" + i + " : ");

      arrayInt[i] = sc.nextInt();

    }

    System.out.print("Msukkan key yang ingin dicari : ");

    key = sc.nextInt();

    for (int i = 0; i < arrayInt.length; i++) {

      if (arrayInt[i] == key) {

        hasil = i;

        break;

      }

    }

    System.out.println("Key ada dalam array pada posisi indeks ke-" + hasil);

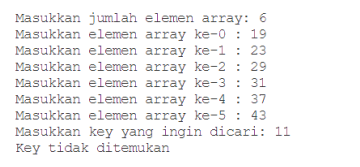
  }

}

3. Modifikasi program pada percobaan 4 di atas, sehingga program akan memberikan pesan

"key tidak ditemukan" jika key tidak ada di dalam array. Contoh tampilan program

sebagai berikut:



**Jawab;**

import java.util.Scanner;

public class LinearSearch14 {

  public static void main(String[] args) {

    Scanner sc = new Scanner(System.in);

    int key = 0;

    int hasil = 0;

    int n\_elemen = 0;

    boolean search = false;

    System.out.print("Masukkan jumlah elemen array : ");

    n\_elemen = sc.nextInt();

    int[] arrayInt = new int[n\_elemen];

    for (int i = 0; i < arrayInt.length; i++) {

      System.out.print("Masukkan elemen array ke -" + i + " : ");

      arrayInt[i] = sc.nextInt();

    }

    System.out.print("Msukkan key yang ingin dicari : ");

    key = sc.nextInt();

    for (int i = 0; i < arrayInt.length; i++) {

      if (arrayInt[i] == key) {

        hasil = i;

        search = true;

        break;

      }

    }

    if (search) {

      System.out.println("Key ada dalam array pada posisi indeks ke-" + hasil);

    } else {

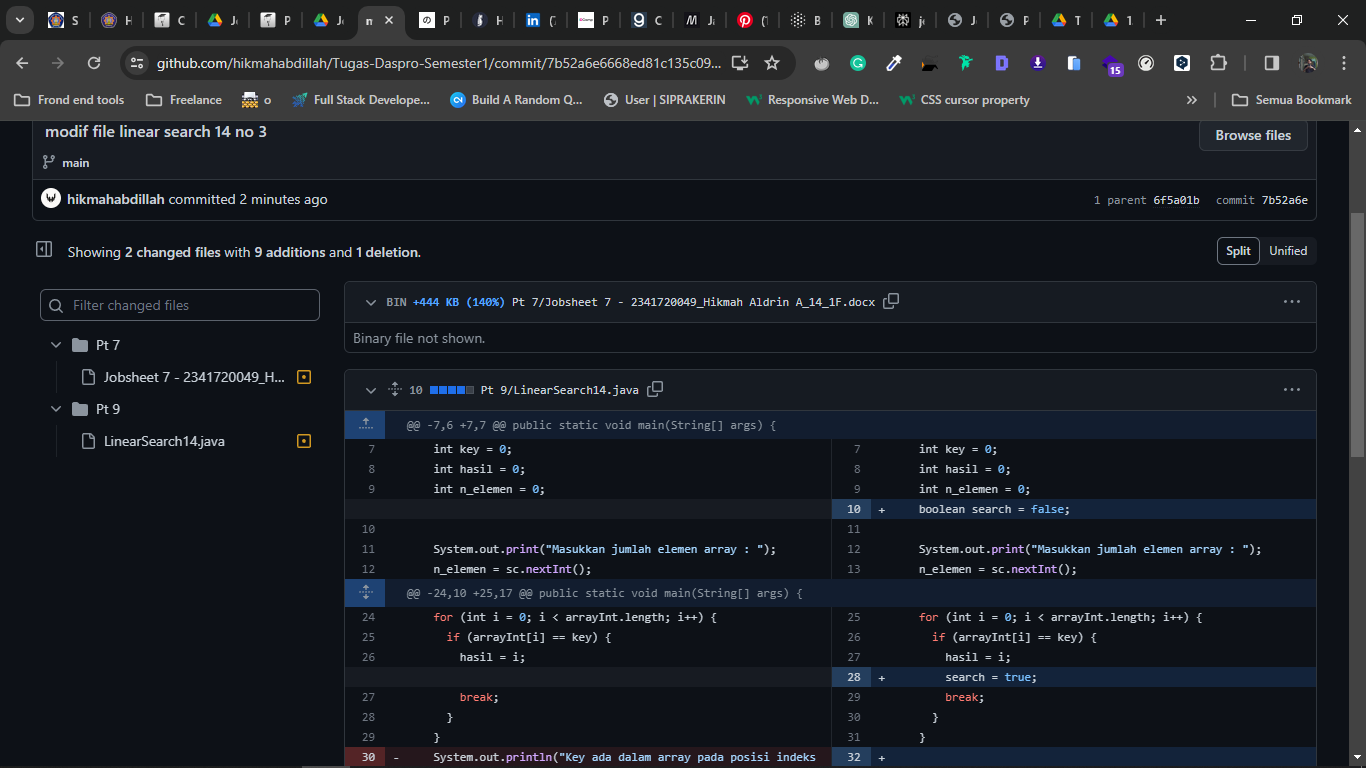
      System.out.println("Key tidak ditemukan dalam array");

    }

  }

}

4. Push dan commit kode program ke github.



**3. Tugas**

1. Buat program untuk menghasilkan nilai tertinggi, nilai terendah, dan rata-rata dari suatu

array berisi bilangan bertipe integer.

Ketentuan:

− Input: Banyaknya elemen, nilai tiap elemen

− Output: Nilai tertinggi, nilai terendah, nilai rata-rata

import java.util.Scanner;

public class TugasArray14 {

  public static void main(String[] args) {

    Scanner sc = new Scanner(System.in);

    // jml elemen

    int x = 0;

    System.out.print("Masukkan jumlah elemen : ");

    x = sc.nextInt();

    double[] nilai = new double[x];

    for (int i = 0; i < nilai.length; i++) {

      System.out.print("Masukkan nilai ke-" + (i + 1) + " : ");

      nilai[i] = sc.nextInt();

    }

    for (int i = 0; i < nilai.length; i++) {

      System.out.println("nilai ke-" + (i + 1) + " adalah " + nilai[i]);

    }

    double max = nilai[0];

    double min = nilai[0];

    double avg = 0;

    for (int i = 0; i < nilai.length; i++) {

      if (nilai[i] > max) {

        max = nilai[i];

      }

      if (nilai[i] < min) {

        min = nilai[i];

      }

      avg += nilai[i];

    }

    avg /= x;

    System.out.println("========================");

    System.out.println("Nilai tertinggi: " + max);

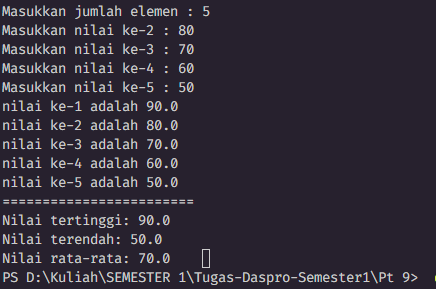
    System.out.println("Nilai terendah: " + min);

    System.out.println("Nilai rata-rata: " + avg);

  }

}

OUTPUT



2. Implementasikan flowchart yang telah dibuat pada tugas pertemuan 9 mata kuliah Dasar

Pemrograman terkait project kelompok ke dalam kode program Java.

Push dan commit hasil kode program Anda ke repository github project Anda.

Catatan: tugas hanya boleh menerapkan materi dari pertemuan 1 hingga pertemuan 9.

**Jawab : -**